Please replace the paragraph appearing at page 18, line 17 to page 19, line 6 with the following:

a2

In one embodiment, once peripheral control code has been downloaded to the peripheral 40, the code is provided to the peripheral a second time. The peripheral 40 utilizes this second copy of the code in a verification procedure, comparing the stored first copy to the newly transmitted second copy. If differences are found between the two versions of the code, then the version of the code which was downloaded and stored is not deemed authentic. The controller 54 of the peripheral 40 may then be arranged to request a new, third copy of the code for download and storage in replacement of the code which is currently stored, and verification procedure may repeat. In this embodiment, the second copy of the code is not stored permanently at the peripheral 40, but is only used in a comparison procedure. As is well known, this comparison procedure may comprise a bit-for-bit comparison or other method of verification now known or later developed. Of course, in this embodiment, the controller 54 of the peripheral 40 is provided with code arranged to cause the peripheral 40 to re-request the code after it has been stored, and to utilize this second requested copy of the code in the verification process.

Please replace the paragraph appearing at page 23, line 19 to page 24, line 3 with the following:

a3

Figure 6 illustrates an operation flow diagram of an example method of creating the authentication file. This method is one exemplary method of operation and it is contemplated that other methods of creating authentication data may be utilized. Further, this method is available for a3

use on any of a removable media, fixed or mass media, software stored on a network, or other any other data storage apparatus. For example, the method is available for authenticating peripheral code which is stored on a removable CD-ROM associated with the master gaming controller 42. The method is also available for authenticating peripheral code which is stored at the mass storage device 46 of the master gaming controller 42.

Please replace the paragraph appearing at page 24, lines 11-20 with the following:

a4

At a step S150 the authentication data creation process loads software application files, such as the peripheral control code or video/audio peripheral operational data to a removable media. In other methods, the software may comprise files other than application files and the files may be loaded on the media prior to the initiation of this process. Next, at a step S152, the operation creates a shell file that will become the authentication file storing the FVT.

Please replace the paragraph appearing at page 24, line 22 to page 25, line 11 with the following:

a5

Thereafter, at a step S158, the operation stores the hash value in the FVT. In one preferred embodiment the hash value is stored with an association with the application file from which the hash value was created. Next, at a decision step S160, the operation determines if there are additional files on the media to execute the hash operation. If there are files for which a hash value has not been created, then the operation returns to step S154 and the operation repeats. If at decision step S160 the operation determines that no additional files exist on which to perform the hash a5

operation, then the operation progresses to a step S162 and the method executes the hash operation on all hash values presently stored in the FVT. The hash operation creates a unique hash value for the hash values stored in the FVT to provide means to detect tampering or unwanted alteration of the hash values in the FVT. This hash value generated by executing the hash operation on the stored hash values is referred to herein as a content signature of the hash values. Next, at a step S164, the operation encrypts the content signature, stores it in the FVT; then, the operation hashes the entire FVT file and obtains a signature for the entire FVT file.

Please replace the paragraph appearing at page 27, lines 12-23 with the following:

a6

Next, at a step S356, the operation searches the media for the verification file stored on the media. The creation and content of the verification file is discussed above. At a step S358, the operation utilizes the decryption algorithms from the secure memory to decrypt the file signature stored in the FVT . The encrypted file signature is shown as element S286 on Figure 7. After decrypting the file signature value stored in the FVT, the operation performs a hash operation on the FVT file up to the encrypted content signature S284 (see Figure 7), to obtain a re-calculated file signature. This occurs at a step S360. Thereafter, at a step S362, the operation compares the decrypted signature to the re-calculate file signature to check for differences in the values. At a decision step S364, a determination is made whether the signatures match. If the decrypted signature does not match the re-calculated signature, the operation progresses to a step S366 and the process terminates. Such a failure to match at step S364 indicates possible tampering or alteration and the installation or game operation should not occur or may have occurred inaccurately.

Please replace the paragraph appearing at page 28, line 20 to page 29, line 2 with the following:

a7

At a step S384, the operation compares the hash value from the FVT to the re-calculated hash value for the corresponding software file stored on the media. At a decision step S386 a determination is made as to whether these two hash values match. If the values do not match, the operation moves to a step S388 and the process terminates. If the values match, the operation moves to a decision step (not shown) wherein the operation determines if all the entries of the FVT have been compared to re-calculated values.

Please replace the paragraph appearing at page 29, lines 4-12 with the following:

a8

If at the decision step there are additional FVT entries to compare, the operation returns to step S380 and the operation repeats as shown. If at the decision step all the FVT entries have been compared to re-calculated entries, the operation progress to a step wherein the determination is made that the media (such as peripheral control code files) has been authenticated. It is contemplated that this process can occur on any media (including control code files, operational data such as audio/video data) for which authentication is desired. It is further contemplated that many other variations may be made to the general process outlined herein without departing in scope of authentication to determine that the software control code on the media, fixed, removable, or otherwise, is trustworthy.